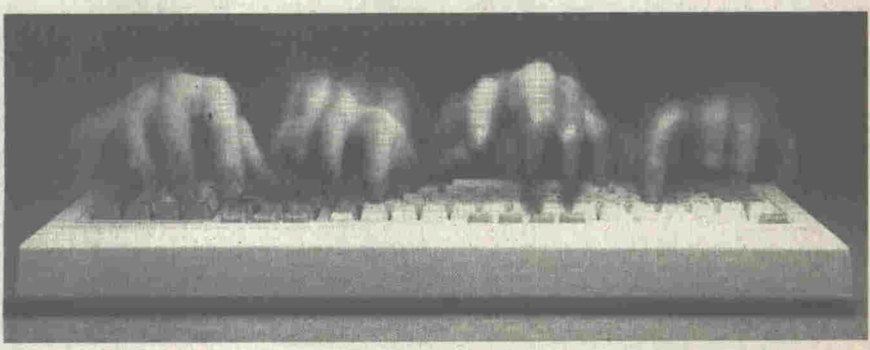
■ โยธิน อินบรรเลง



เขียนโปรแกรมเข้ารหัสด้วย FoxPro

ความลับไม่มีในโลก ถ้อยวลีนี้อาจจะเป็นจริง แต่ว่าเราจะยอมให้บุคคลที่ไม่ได้รับอนุญาตเข้ามาดูข้อมูลของเราได้ง่าย ๆ ได้อย่างไร...

ใน FoxPro เองก็จะมีฟังก์ชันสำหรับเข้ารหัสข้อมูลได้ง่าย ๆ ไม่ยุ่งยากจนเกินไปที่จะทำความเข้าใจ และก็เป็นฟังก์ชันที่ใช้งานได้ดีระดับหนึ่งซึ่งก็พอเพียงสำหรับโปรแกรมที่ใช้อยู่ภายในองค์กร ให้รอดพ้นจากมือดีที่จะแอบเข้ามาดูข้อมูลง่าย ๆ โดยใช้คำสั่ง Browse หรือ Display และ List หรือถ้าจัดวิธีการเข้ารหัสให้ดีรับรองได้ผลเกินคาด

...FoxPro มีฟังก์ชันสำหรับเข้ารหัสข้อมูลได้ง่าย ๆ ซึ่งพอเพียงสำหรับโปรแกรมที่ใช้อยู่ภายในองค์กร ให้รอดพ้นจากมือดีที่จะแอบเข้ามาดูข้อมูล...

การเข้ารหัสข้อมูล

การเข้ารหัสข้อมูลนั้น เราจะใช้ฟังก์ชัน CHRTRANS() ซึ่งจะเป็นฟังก์ชันสำหรับแปลงข้อมูลชุดตัวอักษรโดยอาศัยหลักการการแทนที่โดยอ้างอิงตำแหน่งของตัวอักษรอีกสองชุด

รูปแบบ

```
CHRTRAN(Expression1, Expression2,Expression3)
```

รายละเอียด

Expression1

ชุดตัวอักษรที่เราต้องการเปลี่ยนตัวอักษรข้างใน

Expression2

ชุดตัวอัลษรที่จะใช้ค้นหาในชุดตัวอักษรที่ต้องการเปลี่ยน (Expression1) ซึ่งถ้าหาเจอจะนำเอาตำแหน่งของตัวอักษรใน Expression2 ไปค้นหาใน Expression3 ในตำแหน่งเดียวกัน

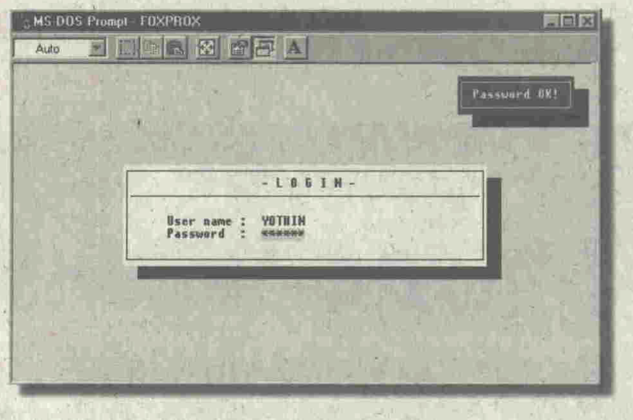
Expression3

จะเป็นชุดตัวอักษรที่ใช้แทนลงในชุดตัวอักษร Expression1 เมื่อตำแหน่งในปัจจุบันของ Expression3 และ Expression2 ตรงกัน

หมายเหตุ

ถ้าข้อมูลใน Express2 มากกว่าตัวอักษรใน Express3 โปรแกรมจะแทนค่าเฉพาะตัวอักษรที่มีเท่านั้น ส่วนที่เหลือจะถูกตัดออกไป แต่ถ้า Express2 มีน้อยกว่า Express3 ข้อมูลตัว

FoxPro



```
Password OK!

- L O G I N -
User name :  YOTHIN
Password  :  ******
```

อักษรที่ไม่มีใน Express2 จะไม่มีการแทนค่าใด ๆ เลย ตัวอย่างเช่น

```
exp1 = 'abc'
exp2 = 'abcdefg'
exp3 = '0123'
x = CHRTRAN(exp1,exp2,exp3)
```

x จะมีค่าเป็น 012 แต่ถ้า exp1 เป็น 'abcfga' ค่าของจะเป็น 0120 จะเห็นได้ว่าตำแหน่งของอักษร e และ f ซึ่งมีค่าเป็น 5 และ 6 ตามลำดับไม่มีใน exp3 ซึ่งมีเพียงตำแหน่งที่ 4 เท่านั้น จะทำให้โปรแกรมไม่สามารถจะเปรียบเทียบได้จึงแทนค่าตัวอักษรด้วยค่าว่าง (NULL) คือ '' ถ้าเอามารวมกับอักษรอื่น ๆ แล้วเราจะเห็นว่าตัวอักษรจะหายไป

การถอดรหัส

หลังจากที่เราเข้ารหัสข้อมูลแล้วเวลาเราจะใช้ข้อมูลนั้น ๆ เราจำเป็นต้องถอดรหัสเสียก่อน ฟังก์ชันที่ใช้ก็เป็นฟังก์ชันเดิมนั่นแหละ แต่เราจะใช้วิธีกลับชุดตัวอักษรที่ใช้แทนค่าก็จะได้ข้อมูลที่สามารถอ่านได้

รูปแบบ

```
CHRTRAN(Expression1, Expression3, Expression2)
```

ส่วนรายละเอียดต่าง ๆ ก็จะเหมือนเดิม

ประกอบเข้าเป็นฟังก์ชันใช้งาน

หลังจากที่เราทราบวิธีการเข้ารหัสและถอดรหัสข้อมูลแล้ว เราก็สามารถนำไปใช้งานได้ทันที แต่การที่เราจะใช้คำสั่งเหล่านี้ซ้ำ ๆ เราก็สร้างให้เป็นฟังก์ชันใหม่ที่สามารถเรียกใช้ได้อย่างสะดวกคือ encode( ) และ decode( )

ฟังก์ชันนี้จะเป็นฟังก์ชันสำหรับเข้ารหัสข้อมูลให้กับข้อมูลที่เรากำหนด ก่อนอื่นเราต้องกำหนดชุดตัวแปรสำหรับเข้ารหัสถอดรหัสเสียก่อนคือ

```
'alpha =
'ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz01234567890 '
key = '๐ขฒงฉซ๒คญฏฐกฑผณดชตฤวธจ๔ูนมฎบฌ๖ปฝฟ๙
ภยพหล๘ศรสพอฮ+๑๒๓฿๕๗ท()?.,/'
```

ชุดตัวแปร key เราจะกำหนดโดยไม่เรียงตัวอักษรเพราะจะทำให้ผู้ที่จะเจาะข้อมูลเดาได้ง่าย ๆ และตัวอักษรว่าง (Space) ก็ให้แทนด้วยตัวอักษรเช่นกันก็จะทำให้แยกแยะไม่ออก

หลังจากนั้นก็ส่งค่าข้อมูลที่เข้ารหัสหรือถอดรหัสกลับ ลองดูรายละเอียดจากฟังก์ชันต่อไปนี้

```
*
* Procedure : encode/decode
* Purpose : ใช้สำหรับเข้ารหัสหรือถอดรหัสข้อมูลตามที่กำหนด
(ในโปรแกรมนี้จะเข้ารหัส
*      เฉพาะภาษาอังกฤษและตัวเลขเท่านั้น)
* Parameter : mCode -> ชุดข้อมูลที่ต้องการเข้ารหัส
* Return : ข้อมูลที่เข้ารหัสข้อมูลเรียบร้อยแล้ว
*
PROC encode  && สำหรับเข้ารหัส
PROC decode  && สำหรับถอดรหัส
PARA mCode
alpha =        'ABCDEFGHIJKLMNOPQRSTUVWXYZ
               abcdefghijklmnopqrstuvwxyz01234567890'
key = '๐ขฒงฉซ๒คญฏฐกฑผณดชตถวธจ๔ูนมฎบฌ๖ปฝฟ๙
     ภยพหล๘ศรสพอฮ+๑๒๓฿๕๗ท()?.,'
RETU CHRTRAN(mCode,alpha,key)
&& สำหรับฟังก์ชัน encode
RETU CHRTRAN(mCode,key,alpha)
&& สำหรับฟังก์ชัน decode
```

โปรแกรมการใช้งาน

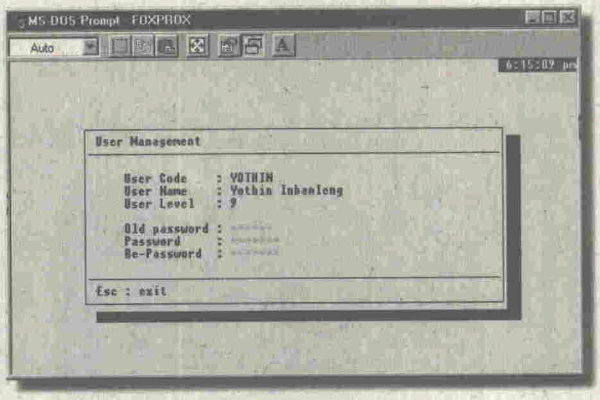
GETPWD()

โปรแกรม GETPWD() ใช้สำหรับรับข้อมูลที่ป้อนเข้ามาทางหน้าจอและแสดงผลทางหน้าจอเป็นเครื่องหมาย * แทน หลังจากนั้นก็เอาข้อมูลที่ได้ไปตรวจสอบข้อมูลในแฟ้มเก็บ Password ที่เข้ารหัสป้องการอ่านจากแฟ้มข้อมูลอยู่แล้วต้องใช้ฟังก์ชัน DeCode() เข้ามาช่วยอีกที

```
*
* Procedure : GetPwd()
* Purpose : ใช้สำหรับรับข้อมูลที่ป้อนจากหน้าจอ
และแสดงผลเป็นเครื่องหมาย * ที่หน้าจอแทน
* Parameter : gRow -> บรรทัดที่ต้องการแสดงเครื่องหมาย *
* gCol -> คอลัมน์ที่ต้องการแสดงเครื่องหมาย *
*      gLen -> ความยาวสูงสุดของ Password ที่รับ
* Return : ข้อความที่ป้อน
*
```

→

FoxPro



```
User Management

    User Code     : YOTHIN
    User Name     : Yothin Inbanleng
    User Level    : 9

    Old password  : ******
    Password      : *******
    Re-Password   : *******

Esc : exit
```

```
* Example : mPwd = GETPWD(5,5,15)
*
PROC          getpwd
PARA          gRow,gCol,gLen
PRIVgRow,gCol,gLen
* ตรวจสอบพารามิเตอร์ที่ส่งมาถ้าไม่ได้ระบุ
ให้กำหนดค่าปกติเป็นกึ่งกลางหน้าจอ
IF TYPE('gLen') <> 'N'
gLen = 15
ENDIF
IF TYPE('gRow') <> 'N'
gRow = WROW()/2
ENDIF
IF TYPE('gCol') <> 'N'
gCol = (WCOL()/2)-gLen
ENDIF
IF TYPE('gColor') <> 'C'
gColor = SET('COLOR')
ENDIF
i = 0
keyGet = ''
@gRow,gCol SAY '' COLO (gColor)
DO WHILE .t.
mGet = INKEY(0)
DO CASE
CASE mGet = 127
keyGet = LEFT(keyGet,LEN(keyGet)-1)
i = i - 1
IF i < 0
i = 0
keyGet = ''
ENDIF
@gRow,gCol SAY keyGet
CASE mGet <> 27 .AND. mGet <> 127 .AND. mGet <> 13
keyGet = keyGet + CHR(mGet)
@gRow,gCol+i SAY '*' COLO (gColor)
i = i+1
IF i = gLen
EXIT
ENDIF
CASE mGet = 13
EXIT
CASE mGet = 27
EXIT
ENDCASE
ENDDO
RETU keyGet
```

การนำไปใช้งาน

```
mPwd = GETPWD(10,15,10)
```

จะได้ตัวแปรที่เก็บข้อมูลที่ป้อนชื่อ mPwd หลังจากนั้นก็นำไปตรวจสอบกับแฟ้มข้อมูล ถ้ามีการเข้ารหัสก็ใช้ฟังก์ชัน EnCode() และ DeCode() เข้ามาช่วย ดังตัวอย่าง

mPwd = EnCode(mPwd) && สำหรับในไปตรวจสอบจากแฟ้มข้อมูลที่เข้ารหัสแล้ว

หรือจะเก็บตัวแปรรหัสในแฟ้มมาแล้วใช้ฟังก์ชัน DeCode() เข้าแปลงอีกทีก็ได้ เช่น

mPass = DeCode(USRPASS) && นำข้อมูลจากฟิลด์ usrpass ไปเก็บว่าที่ mPass พร้อมทั้งถอดรหัส

เห็นหรือยังครับว่า FoxPro ก็มีทีเด็ดดี ๆ ไว้ให้นักพัฒนาได้สร้าง Utility เจ๋ง ๆ ได้เหมือนกัน ส่วนท่านไหนต้องการโปรแกรมจัดการ User แบบเต็ม ๆ ก็ใช้บริการซอฟต์เมล์ได้นะครับ และถ้าต้องการทราบเรื่องอะไรเกี่ยวกับ FoxPro ก็ติดต่อทาง yothinin@hotmail.com ยินดีต้อนรับเสมอ